

TUGAS PEKAN 7
ALGORITMA DAN PEMROGRAMAN
SISTEM REGISTRASI KREDENSIAL PENGGUNA

Disusun oleh:

Nama : Thaariq Salam

NIM : 2511532022

Kelas : Informatika / B

Dosen Pengampu: Dr. Wahyudi S.T.M.T

Asisten Praktikum: Rahmad Dwirizki Olders



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
TAHUN 2025

DAFTAR ISI

DAFTAR ISI	i
1. Soal	1
2. <i>Pseudocode</i>	1
3. <i>Flowchart</i>	4
4. <i>Source Code</i>	5
5. <i>Screenshoot Output</i>	6
6. Penjelasan Singkat Program	7
1. Class Data (<i>Akun.java</i>)	7
2. Class Utama (<i>tugasAlproPekan7_2511532022.java</i>)	8

1. Soal

Sistem Registrasi Kredensial Pengguna

Buatlah program untuk mensimulasikan proses registrasi akun baru. Program harus dapat mengambil input pengguna, menyimpannya dalam sebuah *object*, dan melakukan validasi data berdasarkan aturan yang ditentukan sebelum menampilkan hasilnya.

Struktur Program:

1. Class Data: Akun.java
 - Class ini bertugas sebagai *blueprint* untuk menyimpan data akun.
 - Atribut yang harus ada: username, password, email (semua String), dan pinAngka (int).
 - Gunakan setter dan getter jika dibutuhkan untuk mengelola data privat tersebut.
 - Wajib membuat *helper method* validasi berikut di dalam *class* ini:
 - a. public boolean isPasswordValid(): Mengembalikan true jika panjang password 8 karakter atau lebih.
 - b. public boolean isEmailValid(): Mengembalikan true jika email mengandung karakter @ dan "."
2. Class Utama: Registrasi.java
 - Class ini memiliki main method.
 - Bertugas mengambil input, membuat *object* Akun, dan memanggil *setter*.
 - Meminta input berikut dari pengguna:
 - a. Username (String)
 - b. Password (String)
 - c. Email (String)
 - d. PIN Angka (int, misal 6 digit)
 - Melakukan pengecekan if-else dengan memanggil method isPasswordValid() dan isEmailValid() dari *object* Akun.

Gambar 1.1

2. Pseudocode

A. Judul

Judul program : Sistem Registrasi Kredensial Pengguna
Nama file utama : tugasAlproPekan7_2511532022.java
Nama file data : Akun.java

B. Deklarasi :

Komponen	Tipe data	Keterangan
Akun.java	Class	Template untuk Objek Akun
username, password, email	String	Atribut data dalam Class Akun
pinAngka	Integer	Atribut data dalam <i>Class Akun</i>
scanner	Scanner Object	Objek yang menerima data registrasi pengguna
objectAkunBaru	Akun Object	Objek yang menyimpan data registrasi pengguna
isUserOK, isPassOK, isEmailOK, isPinOK	Boolean	Variabel untuk menyimpan hasil validasi (True/False)
Setter dan Getter	Method	Semua method set.. dan get.. untuk semua atribut di <i>Class Akun</i> (<i>username, password, email, pinAngka</i>)
Helper Method Validasi	Method	Semua method is...Valid di Class Akun

Tabel 2.1

C. Pseudocode

1. Class data (Akun.java)

a. Konstruktor (private) akun

- 1) Mulai prosedur
- 2) Inisialisasi `this.username ← username.` dan yang lainnya
- 3) End prosedur

b. Setter dan Getter : set. ... dan get. .. untuk username, password, email, pinAngka.

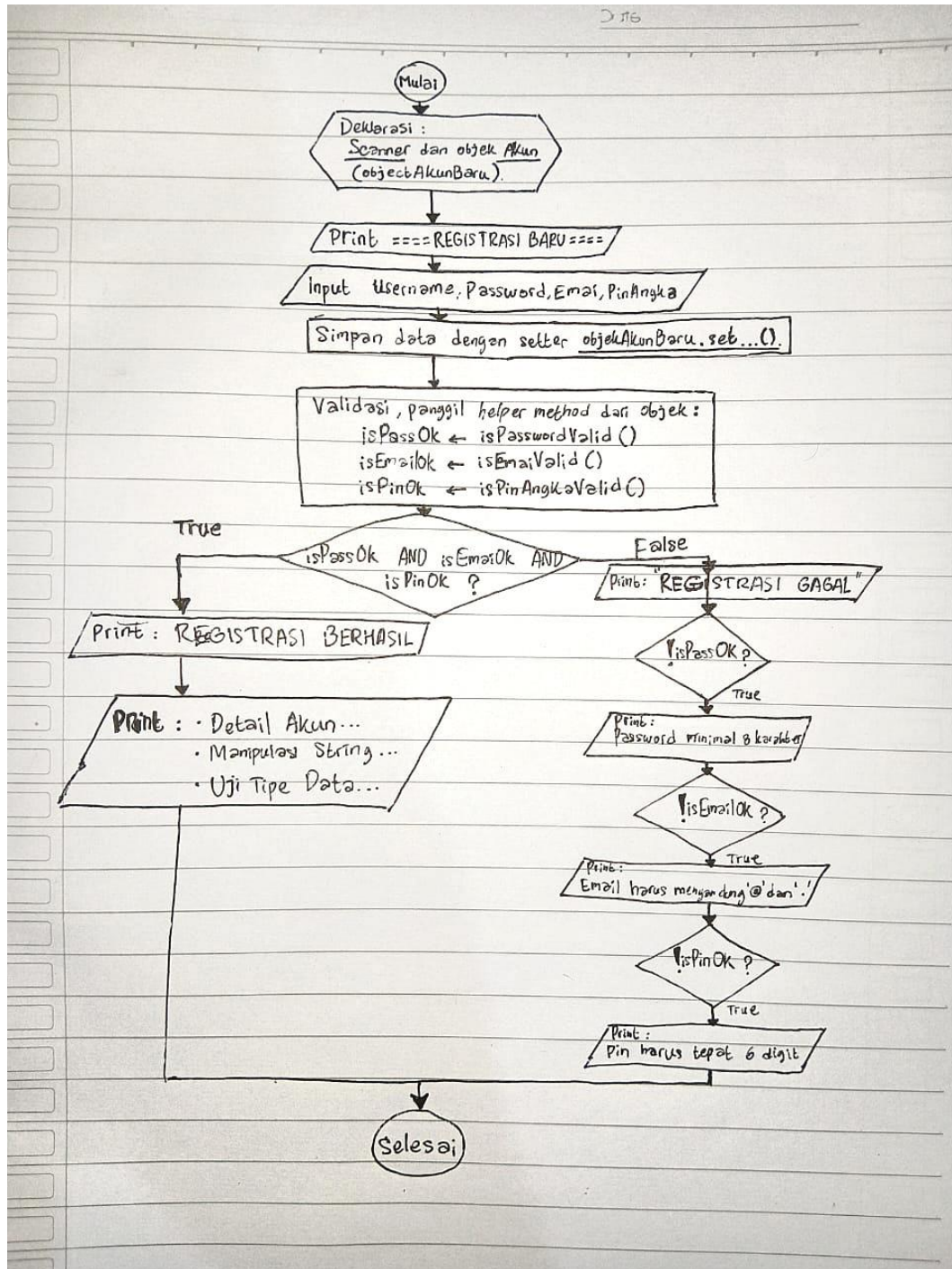
c. Validasi

- 1) Mulai validasi password, email, pinAngka
- 2) Return panjang password ≥ 8
- 3) ditemukanAt \leftarrow this.email indexOf '@' $\neq -1$
- 4) ditemukanTitik \leftarrow this.email indexOf '.' $\neq -1$
- 5) Return ditemukanTitik && ditemukanAt
- 6) Return panjang (koversi this.pinAngka ke String) ==6

2. Class Utama

- 1) Mulai
- 2) Deklarasi *scanner* dan *objectAkunBaru*
- 3) Print ===== REGISTRASI AKUN BARU =====
- 4) Input data (*setter*)
 - baca input username, panggil *objectAkunBaru.setUsername*
 - baca input password, panggil *objectAkunBaru.setPassword*
 - baca input email, panggil *objectAkunBaru.setEmail*
 - baca input pinAngka, panggil *objectAkunBaru.setPinAngka*
- 5) *Close scanner*
- 6) Periksa validasi *input* dengan nilai boolean
- 7) *isPassOK* ← panggil *objectAkunBaru.isPasswordValid()*
- 8) *isEmailOK* ← panggil *objectAkunBaru.isEmailValid()*
- 9) *isPinOK* ← panggil *objectAkunBaru.isPinAngkaValid()*
- 10) IF (*isUserOK* AND *isPassOK* AND *isEmailOK* AND *isPinOK*) THEN
 - a. Print "REGISTRASI BERHASIL"
 - b. Print "Akun untuk "" + *objectAkunBaru.getUsername()* + ""
telah berhasil dibuat."
 - c. Print "--- Detail Akun---"
 - d. Print Username dengan *.toLowerCase()*
 - e. Print Email dengan *.toUpperCase()*
 - f. Print ID Gabungan dengan *.concat()*
 - g. Print Hasil Uji Tipe Data: PIN (int) + 10
 - h. Print Hasil Uji Tipe Data: PIN (String) + "10"
- 11) END IF
- 12) ELSE
- 13) Print "REGISTRASI GAGAL" dan "Alasan Gagal:"
- 14) IF *isPassOK* False, THEN Print pesan error Password. END IF
- 15) IF *isEmailOK* False, THEN Print pesan error Email. END IF
- 16) IF *isPinOK* False, THEN Print pesan error PIN. END IF
- 17) Selesai

3. Flowchart



Gambar 3.1

4. Source Code

Source code terdiri dari 2 buah *class* yaitu: gambar 4.1 adalah *class* data (Akun.java) dan gambar 4.2 adalah *class* utama/driver (tugasAlproPekan7_2511532022.java).

```
package tugasAlproPekan7_2511532022;

public class Akun {
    private String username;
    private String password;
    private String email;
    private int pinAngka;

    public Akun ( String username, String password, String email, int pinAngka) {
        this.username = username;
        this.password = password;
        this.email = email;
        this.pinAngka = pinAngka;
    }

    // ini SETTER ya
    public void setUsername (String username) {
        this.username = username;
    }
    public void setPassword (String password) {
        this.password = password;
    }
    public void setEmail (String email) {
        this.email = email;
    }
    public void setPinAngka (int pinAngka) {
        this.pinAngka = pinAngka;
    }

    // nah ini GETTER nya ya
    public String getUsername() {
        return username;
    }
    public String getPassword() {
        return password;
    }
    public String getEmail() {
        return email;
    }
    public int getPinAngka() {
        return pinAngka;
    }

    // ini HELPER METHOD VALIDATION nya ,OK
    public boolean isPasswordValid() {
        return this.password.length() >= 8;
    }
    public boolean isEmailValid() {
        return this.email.indexOf('@') != -1 && this.email.indexOf('.') != -1;
    }
    public boolean isPinAngkaValid() {
        return String.valueOf(this.pinAngka).length()==6;
    }
}
```

Gambar 4.1

```

package tugasAlproPekan7_2511532022;
import java.util.Scanner;
public class tugasAlproPekan7_2511532022 {
    public static void main (String[] args) {
        Scanner scanner = new Scanner(System.in);
        Akun objectAkunBaru = new Akun("", "", "", 0);

        System.out.println("==== REGISTRASI AKUN BARU =====");
        System.out.print("Masukkan Username: ");
        objectAkunBaru.setUsername (scanner.nextLine());
        System.out.print("Masukkan Password: ");
        objectAkunBaru.setPassword (scanner.nextLine());
        System.out.print("Masukkan Email: ");
        objectAkunBaru.setEmail (scanner.nextLine());
        System.out.print("Masukkan PIN (6 angka): ");
        objectAkunBaru.setPinAngka (scanner.nextInt());
        scanner.close();

// panggil semua HELPER METHOD
        boolean isPassOk= objectAkunBaru.isPasswordValid();
        boolean isEmailOk= objectAkunBaru.isEmailValid();
        boolean isPinOk= objectAkunBaru.isPinAngkaValid();
        int y =10;
        String z ="10";

        if (isPassOk && isEmailOk && isPinOk) {
            System.out.println("\n--- REGISTRASI AKUN BERHASIL --- ");
            System.out.println("Akun untuk \"" + objectAkunBaru.getUsername() + "\" berhasil dibuat.");

            System.out.println("\n--- DETAIL AKUN --- ");
            System.out.println("Username (Lowercase)\t: " + objectAkunBaru.getUsername().toLowerCase());
            System.out.println("Email (Uppercase)\t: " + objectAkunBaru.getEmail().toUpperCase());
            System.out.println("ID Pengguna (Gabungan)\t: " + objectAkunBaru.getUsername().concat(String.valueOf(objectAkunBaru.getPinAngka())));

            System.out.println("\n--- Uji Tipe Data (PIN Anda: " + objectAkunBaru.getPinAngka() + ") ---");
            System.out.print("PIN (int) + 10 \t\t= ");
            System.out.println(objectAkunBaru.getPinAngka()+ y);
            System.out.print("PIN (String) + 10 \t= ");
            System.out.println(objectAkunBaru.getPinAngka() + z );

        } else {
            System.out.println("\n--- REGISTRASI GAGAL ---");
            if (!isPassOk) {
                System.out.println("Password Anda \"" + objectAkunBaru.getPassword() + "\" tidak valid (Password minimal 8 karakter).");
            }
            if (!isEmailOk) {
                System.out.println("Email Anda \"" + objectAkunBaru.getEmail() + "\" tidak valid (harus mengandung '@' dan '.').");
            }
            if (!isPinOk) {
                System.out.println("PIN Anda \"" + objectAkunBaru.getPinAngka() + "\" tidak valid (harus tepat 6 angka)");
            }
            System.out.println("Silahkan coba lagi ");
        }
    }
}

```

Gambar 4.2

5. Screenshoot Output

- Jika validasi berhasil:

```

===== REGISTRASI AKUN BARU =====
Masukkan Username: fufufafa67
Masukkan Password: 6767676767
Masukkan Email: fufufafa67@wapres.paman
Masukkan PIN (6 angka): 676767

--- REGISTRASI AKUN BERHASIL ---
Akun untuk "fufufafa67" berhasil dibuat.

--- DETAIL AKUN ---
Username (Lowercase)      : fufufafa67
Email (Uppercase)         : FUFUFAFA67@WAPRES.PAMAN
ID Pengguna (Gabungan)    : fufufafa67676767

--- Uji Tipe Data (PIN Anda: 676767) ---
PIN (int) + 10             = 676777
PIN (String) + 10          = 67676710

```

Gambar 5.1

- Jika validasi gagal:

```
===== REGISTRASI AKUN BARU =====
Masukkan Username: fufufafa67
Masukkan Password: 6767676767
Masukkan Email: fufufafa67
Masukkan PIN (6 angka): 676767

--- REGISTRASI GAGAL ---
Email Anda "fufufafa67" tidak valid (harus mengandung '@' dan '.').
Silahkan coba lagi
```

Gambar 5.2

```
===== REGISTRASI AKUN BARU =====
Masukkan Username: fufufafa67
Masukkan Password: 676767
Masukkan Email: fufufafa67
Masukkan PIN (6 angka): 6767

--- REGISTRASI GAGAL ---
Password Anda "676767" tidak valid (Password minimal 8 karakter).
Email Anda "fufufafa67" tidak valid (harus mengandung '@' dan '.').
PIN Anda "6767" tidak valid (harus tepat 6 angka)
Silahkan coba lagi
```

Gambar 5.3

6. Penjelasan Singkat Program

Program ini dibagi menjadi dua *class* utama untuk menerapkan prinsip pemisahan tanggung jawab dan tugas.

1. Class Data (*Akun.java*)

Bertindak sebagai cetak biru atau wadah data. *Class* ini bertanggung jawab penuh atas struktur dan integritas data akun.

- *Encapsulation*: Semua data sensitif (*username*, *password*, *email*, *pinAngka*) dideklarasikan sebagai *private* dan hanya dapat diakses melalui *method* publik (*public*) yaitu *getter* (*get...*) dan *setter* (*set...*).
- *Validasi (Helper Method)*: Logika aturan diletakkan di sini dalam bentuk *helper method* (*isUsernameValid()*, *isPasswordValid()*, dll.). *Class* ini

memastikan data memenuhi syarat (misalnya, password minimal 8 karakter atau PIN harus 6 digit) sebelum digunakan oleh *class* lain.

2. Class Utama (*tugasAlproPekan7_2511532022.java*)

Bertindak sebagai *Driver* atau Pengendali alur program.

Alur Logikanya :

- Menggunakan objek *Scanner* untuk mengambil semua *input* data dari pengguna.
- Membuat objek *Akun* (*objectAkunBaru*) dan mengisi datanya menggunakan *Setter*.
- Percabangan *if-else* digunakan untuk memanggil semua *helper method* validasi dari objek *Akun*. Jika semua validasi mengembalikan nilai *True*, maka registrasi berhasil.
- Melakukan semua operasi *string* (*.toLowerCase()*, *.toUpperCase()*, *.concat()*) dan uji tipe data.
- Jika ada yang bernilai *false* maka akan muncul pesan bahwa *input*-an belum valid.

Dengan pemisahan tugas ini, program menjadi modular. *Class* Utama hanya perlu bertanya kepada objek *Akun* ("Apakah *input*-an valid?") dan objek *Akun* akan menjawabnya berdasarkan aturan yang dimilikinya.